43

EXPRESS MAIL NO. EL896636742US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Siani Lynne Pearson, et al.)
)
Serial No.: Not yet assigned)
)
Filed: concurrently herewith) Our Ref: B-4279 619006-5
)
For: "TRUSTED SYSTEM") Date: August 17, 2001

CLAIM TO PRIORITY UNDER 35 U.S.C. 119

Commissioner of Patents and Trademarks
Box New Patent Application
Washington, D.C. 20231

Sir:

[X] Applicants hereby make a right of priority claim under 35 U.S.C. 119 for the benefit of the filing date(s) of the following corresponding foreign application(s):

COUNTRY	FILING DATE	SERIAL NUMBER
Great Britain	18 August 2000	0020416.4

[ ] A certified copy of each of the above-noted patent applications was filed with the Parent Application No._____.

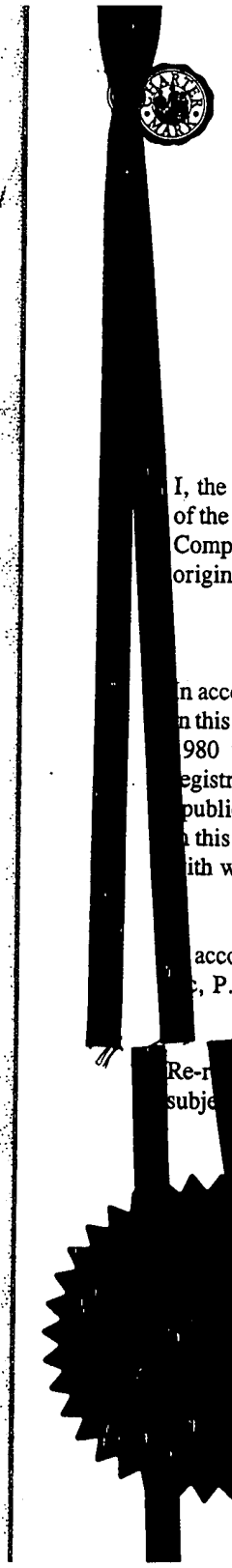
[X] To support applicants' claim, a certified copy of the above-identified foreign patent application is enclosed herewith.

[ ] The priority document will be forwarded to the Patent Office when required or prior to issuance.

Respectfully submitted,

Richard P. Berg
Attorney for Applicant
Reg. No. 28,145

LADAS & PARRY
5670 Wilshire Boulevard
Suite 2100
Los Angeles, CA 90036
Telephone: (323) 934-2300
Telefax: (323) 934-0202



The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

CERTIFIED COPY OF PRIORITY DOCUMENT

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed M Cook

Dated 1 November 2000

Patents Form 1/77

Patents Act 1977
(Rule 16)

THE PATENT OFFICE
18 AUG 2000
RECEIVED

The Patent Office

1/77

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
South Wales
NP10 8QQ

1. Your reference — 30006639 GB

2. P: — 0020416.4
(The Patent Office will fill in this part)

21AUG00 E562073-1 D01463
P01/7700 0.00-0020416.4

18 AUG 2000

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

Hewlett-Packard Company
3000 Hanover Street
Palo Alto
CA 94304, USA

Patents ADP number *(if you know it)*

Delaware, USA

If the applicant is a corporate body, give the country/state of its incorporation

4965880004

4. Title of the invention — Trusted System

5. Name of your agent *(if you have one)*

"Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

Christopher John Harrison
Hewlett-Packard Ltd, IP Section
Filton Road
Stoke Gifford
Bristol BS34 8QZ

Patents ADP number *(if you know it)*

7963713001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*

a) any applicant named in part 3 is not an inventor, or
b) there is an inventor who is not named as an applicant, or
c) any named applicant is a corporate body.
See note (d))

Yes

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	21
Claim(s)	2
Abstract	1
Drawing(s)	4+4

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	
Request for preliminary examination and search *(Patents Form 9/77)*	1
Request for substantive examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	Fee Sheet

11. I/We request the grant of a patent on the basis of this application.

Signature Richard A Lawrence Date 18/08/00

12. Name and daytime telephone number of person to contact in the United Kingdom: Janet Smith, 0117-312-8026

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) *If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.*

b) *Write your answers in capital letters using black ink or you may type them.*

c) *If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.*

d) *If you have answered 'Yes' Patents Form 7/77 will need to be filed.*

e) *Once you have filled in the form you must remember to sign and date it.*

f) *For details of the fee and ways to pay please contact the Patent Office.*

TRUSTED SYSTEM

Technical Field

The present invention relates to payment transactions.

Background Art

Point-of-sale payment terminals are physical checkout devices currently used for credit, debit and smart card transactions, typically used in shops and small businesses and mainly owned by merchants and banks. These devices capture payment information at the point of sale and quickly transfer it from the merchant counter to the payment network for approval. An example provider is HP VeriFone. Current products enable support for multiple applications or services - such as loyalty programs, payment, smart card processing- at the point-of-sale. Furthermore, multiple applications, created by different developers, can reside on one terminal and yet remain separate. Various handheld and countertop peripheral products support multiple options for secure PINpad and smart card applications at the point of sale. These products allow merchants to accept both debit and smart card forms of payment.

Typically the architecture and functionality of such payment terminals is very different from computer apparatus such as PCs, since the functionality is very specific and the cost must normally be kept very low. However, various types of software attack are possible on these systems, and in addition there is a danger that the merchant may cheat the customer out of money by putting through too much money or putting through a transaction twice.

Summary of the Invention

In accordance with a first aspect of the present invention there is provided a method for allowing a financial transaction to be performed using a electronic system, the method comprising interrogating an electronic transaction terminal with an electronic security device to obtain an integrity metric for the electronic financial transaction terminal; determining if the transaction terminal is a trusted terminal based upon the integrity metric; allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

Preferably the method further comprises the providing of user identification data for the user of the electronic security data to the transaction terminal via the security device to allow authorisation of the transaction associated with the financial transaction data.

In accordance with a second aspect of the present invention there is provided a financial transaction system comprising an electronic financial terminal; an electronic security device having interrogation means for interrogating the electronic financial transaction terminal to obtain an integrity metric for the electronic financial transaction terminal, determining means for determining if the transaction terminal is a trusted terminal based upon the integrity metric, means for allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

In accordance with a third aspect of the present invention there is provided an electronic security transaction device having interrogation means for interrogating an electronic financial transaction terminal to obtain an integrity metric for the electronic financial transaction terminal, determining means for determining if the transaction terminal is a trusted terminal based upon the integrity metric, means for allowing financial transaction

data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

This invention relates to a secure payment transaction method used by customers in establishing trustworthiness of the payment procedure when entering into a transaction via a payment terminal by means of the integrity checking of functional components in this terminal, trusted feedback to the customer and a secure payment protocol.

The invention applies to all types of payment terminal, including countertop, portable or wireless payment terminals.

The present invention adds trusted functionality to payment terminals in order to enhance the trustworthiness of the payment terminals and allow a user to check whether the transaction operation and payment is made in the expected manner.

This invention seeks to provide the user with increased trust and confidence in the payment transaction operation by means of defining a trusted transaction payment protocol and being able to check that this trusted transaction payment protocol is carried out.

Preferably the payment terminal can be trusted by means of mutual authentication and a secure token or trusted personal device of the user carrying out an integrity check on the payment terminal. Either the result of this check is implicit, and the protocol will only be allowed to continue if the token or trusted personal device is satisfied as to the payment terminal's integrity, or the result of the check can be explicit, whereby the result of this check will be displayed on the trusted personal device or else a user's secret image will be displayed on the payment terminal itself. In the latter case the

payment terminal must delete this secret once the transaction is complete, and part of the integrity check on the payment terminal should ensure that the terminal is configured for this to take place. If the result of the check is explicit, the user will only continue with the transaction payment if they are satisfied as to the trustworthiness of the payment terminal. Optionally, any result displayed to the user can include information relating to the trustworthiness of the bank.

Preferably the user's token or trusted personal device displays an image on the payment terminal with another special secret stored within the token/trusted personal device and previously unknown to the payment terminal, or else by displaying directly onto the trusted personal device.

Preferably user authorisation for continuing the procedure of purchasing goods is by means of a hardware switch or software button that the consumer must press. The payment will be made once this button/switch is pressed. The software button could be associated with the image on the payment terminal as described in the second aspect, or else could be displayed on the trusted personal device.

Preferably compartmentalisation within the payment terminal is used to separate different types of transaction, such as different customers' transactions, different types of transaction (e.g. smart card, swipe card and debit card) or different banks.

Preferably compartmentalisation within the trusted personal device is used to separate different types of communication.

Preferably the electronic security device is a wireless trusted personal device on which the various images referred to in the previous aspects are displayed

so that the consumer does not have to be in the same location as the payment terminal and therefore does not have to make payments at fixed points.

The invention provides the advantage of allowing a customer to be able to trust that the payment operation can be trusted; that is to say that the payment operation will be carried out in the expected manner. This involves the customer needing to trust that the terminal itself is operating in the expected manner, that the bank is trustworthy, that the amount paid by the customer will be the amount that the customer expects to be charged, that the customer is buying the goods s/he expects, and so on.

A prior patent application [application ref. 30990050 Trusted Platform] described the use of a Trusted Component ('TC') to enable verification of the integrity of a computer platform by the reliable measurement and reliable reporting of integrity metrics. This enables the verification of the integrity of a platform by either a local user or a remote entity. That prior patent application described a general method of reporting integrity metrics and verifying the correctness of the integrity of a platform by comparing reported values of metrics with proper values of metrics. A further patent application described a method of displaying images sent to the trusted platform such that the image appearing on the monitor can be trusted to correspond to the data input to the client.

Description of the Preferred Embodiment

While the ideas of the invention are general, for ease of discussion, we will focus on preferred embodiments, wherein smart cards are the security tokens held by consumers. In fact, such tokens could be smart cards, other security tokens, secure pinpads, trusted PDAs or other trusted mobile computing

apparatus. Optionally, these devices could themselves be trusted computing platforms (containing a 'TC').

The consumers interact with a trusted checkout box acting as a trusted vendor during the registration and payment procedures. The following is an example of a scenario:

A trusted checkout box is based in a shop. A consumer wishes to buy some goods, which are supplied by the vendor. To make a payment, the consumer is asked to insert his smart card into the smart card reader. After the consumer does this, an image with a special seal generated by the smart card and previously unknown to the checkout box is displayed on the screen, confirming to the consumer that the smart card is satisfied that the checkout box can be trusted. Optionally, this special image can further confirm to the consumer that the remote bank platform, which takes part in this payment process, can be trusted as well. Note that the techniques of the trusted computing platform, integrity check function and special image seal verification have been included in prior patent applications.

After inputting the details or code of the goods, an image with another special seal, again generated by the smart card and previously unknown to the checkout box, is displayed on the screen, confirming to the consumer that the smart card knows the price and product information. Associated with this image is a button, probably a hardware switch that the consumer must press in order to authorise continuing the procedure of purchasing goods. In response to pressing the button, the payment is done.

In this preferred embodiment, there are four entities involved in the procedure of purchasing goods. They are an off-line certificate authority (CA), a consumer with a smart card (SC, for which the name is User), a local

checkout box with a trusted component (TC1), and a remote bank platform with a trusted component (TC2). The architecture and functionality of the trusted component have been disclosed in prior patent applications.

For the purposes of authentication and key distribution, each entity has the following asymmetric key pairs: the CA has a RSA key pair for signature and verification, the SC has a RSA key pair for signature and verification and each of TC1 and TC3 has at least a RSA key pair for signature and verification, or optionally, has two RSA key pairs respectively for signature-verification and encryption-decryption.

The following assumptions are made for trust relationships among the above entities:

- SC, TC1 and TC2 know the public key of CA, and they believe that only CA knows the corresponding private key and this key pair is valid.
- TC1 and TC2 hold certificates of their own public keys issued by CA.
- SC holds a certificate of its own public key either issued by TC2 whose corresponding certificate is issued by CA or (optionally) issued by CA directly.
- SC, TC1 and TC2 believe that any of the other entities that know a private key corresponding to a certificate issued by CA, would follow the protocol properly.

A preferred protocol for implementing the preferred embodiment of the present invention is described blow. In the protocol, the following notations will be used:

- CMD_n – the n_{th} command which is used to indicate different services, probably including the product description, service type, price, payment method, etc.
- $N_{n\text{-}X}$ – the n_{th} nonce generated by the entity X (X = {SC, TC1, TC2});

- $S_x(m)$ – a signature on a data element m signed with a private signature key of the entity X (X = {SC, TC1, TC2});
- $E_x(m)$ – a data element m encrypted by using the public encryption key of the entity X (X = {TC1, TC2});
- *A → B: m* – a data element m is transferred from entity A (A = {SC, TC1, TC2}) to entity B (B = {SC, TC1, TC2});
- *m1,m2* – a concatenation of two data elements m1 and m2;
- *Cert(X)* – a certificate of the entity X's public key issued by the CA, where X = {SC, TC1, TC2}; and
- IM_X – the integrity metrics of the platform with X (X = {TC1, TC2}). The technique of integrity check of a trusted computing platform was disclosed in a prior patent application.

The protocol

This protocol includes the security mechanisms of authentication amongst SC, TC1 and TC2, integrity checking of the checkout box with TC1 and the remote bank platform with TC2, and establishment of a transaction of the payment.

The consumer inserts the smart card in the checkout box to make a purchase request, then:

1. *TC1 → SC:* CMD_1, $N_{1\text{-}TC1}$, *Cert(TC1)*

TC1 initiates this part of the protocol by sending SC a command CMD_1, a newly generated nonce $N_{1\text{-}TC1}$, and its certificate *Cert(TC1)* (if SC does not have this certificate yet).

2. *SC → TC1:* $N_{2\text{-}SC}$,

TC2,

Cert(SC)

Upon receipt of Message 1, SC replies TC1 with another newly generated nonce $N_{2\text{-}SC}$, the name of TC2 and its certificate *Cert(SC)* (if TC1 does not have this certificate yet). After receiving Message 2, the checkout box connects to the remote bank platform, then

3. *TC2 → TC1:* $N_{3\text{-}TC2}$,

Cert(TC2)

TC2 replies TC1 with a newly generated nonce $N_{3\text{-}TC2}$ and its certificate *Cert(TC2)* (if TC1 hasn't got this certificate yet).

4. *TC1 → TC2:* CMD_1,

$N_{2\text{-}SC}$,

Cert(SC),

Cert(TC1)

To reply Message 3, TC1 sends TC2 the command CMD_1, the nonce $N_{2\text{-}SC}$ and a certificate *Cert(SC)* forwarded from SC's message and its own certificate *Cert(TC1)* (if TC2 does not have this certificate).

5. *TC2 → TC1:* IM_{TC2},

$S_{TC2}(CMD_1, N_{2\text{-}SC}, N_{3\text{-}TC2}, User, TC1, IM_{TC2})$

Upon receipt of Message 4, TC2 sends Message 5 to TC1 with the integrity metric of the remote bank platform IM_{TC2} and a signature $S_{TC2}(CMD_1, N_{2\text{-}SC}, N_{3\text{-}TC2}, User, TC1, IM_{TC2})$.

6. *TC1 → SC:* $N_{3\text{-}TC2}$,

IM_{TC1},

IM_{TC2},

Cert(TC2),

$S_{TC1}(CMD_1, N_{2\text{-}SC}, N_{1\text{-}TC1}, User, TC2, IM_{TC1})$,

$S_{TC2}(CMD_1, N_{2\text{-}SC}, N_{3\text{-}TC2}, User, TC1, IM_{TC2})$

After receiving Message 5, TC1 sends Message 6 to SC.

7. $SC \rightarrow TC1$: $S_{SC}(CMD_1, N_{1\text{-}TC1}, N_{3\text{-}TC2}, N_{2\text{-}SC}, TC1, TC2,$

$E_{TC1}(TID, SK1), E_{TC2}(SK2))$

Upon receipt of Message 6, SC verifies both signatures signed by TC1 and TC2 for two purposes of authentication and integrity check. If the verification succeeds, SC makes a signature $S_{SC}(CMD_1, N_{1\text{-}TC1}, N_{3\text{-}TC2}, N_{2\text{-}SC}, TC1, TC2)$ including all the nonces being used in this session and two encrypted data respectively for TC1 and TC2. SC sends this signature to TC1.

8. $TC1 \rightarrow TC2$: $S_{SC}(CMD_1, N_{1\text{-}TC1}, N_{3\text{-}TC2}, N_{2\text{-}SC}, TC1, TC2)$

After receiving Message 7, TC1 forwards the signature to TC2. Then both TC1 and TC2 will verify SC's signature. If this part of the protocol succeeds, TC2 will take the payment.

Note that during the flow of this protocol, if any verification or check is not successful, the corresponding verifier will make an announcement to let the other entities know what happens and then the protocol aborts.

If the information of the payment transaction is sensitive to any other party, the communications between TC1 and TC2 must be protected, for example by using an encrypted channel. In this case, TC1 and TC2 can use their RSA encryption-decryption key pairs to establish an authenticated shared session key, and then use this session key to protect all message flows between them.

Optionally, such technology can be incorporated with wireless technology such as Bluetooth (a wireless transmitter/ receiver programmed to allow a free flow of data without bulky cables, and designed to work anywhere). Using the protocol above with a (long-distance) wireless personal device instead of a

smart card or connected personal device, transactions (payments) are brought to the consumer instead of the consumer having to make payments at fixed points.

5 Incorporated with this application is an annex A – Trusted Computing Platform.

Trusted Computing Platform

(HP Ref 30990050)

Technical Field

The present invention generally relates to trusted devices, trusted computing platforms, trusted transactions and methods of operating the same.

Background Art

For commercial applications, a client computing platform typically operates in an environment where its behaviour is vulnerable to modification by local or remote entities. This potential insecurity of the platform is a limitation on its use by local parties who might otherwise be willing to use the platform, or remote parties who might otherwise communicate with the platform; for example, for the purposes of E-commerce. For the present purposes, both local parties and remote parties will be referred to as "users" unless otherwise stated.

Existing security applications, for example virus detection software, execute on computing platforms under the assumption that the platform will operate as intended and that the platform will not subvert processes and applications. This is a valid assumption provided that the intended software state has not become unstable or has not been damaged by other software such as viruses. Users, therefore, typically restrict the use of such platforms to non-critical applications, and weigh the convenience of the using the platforms against the risk to sensitive or business critical data.

Increasing the level of trust in platforms therefore enables greater user confidence in existing security applications (such as the 'Secure Sockets Layer' or 'IPSec') or remote management applications. This enables greater reliance on those applications and hence reduced 'cost of ownership'. Greater trust also enables new electronic methods of business, since there is greater confidence in the correct operation of both local and remote computing platforms.

In this document, the word 'trust' is used in the sense that something can be 'trusted' if it always behaves in the way it is expected to behave.

Disclosure of the Invention

The present inventors have appreciated that it is desirable to use a physical device in a computing platform to verify and possibly enforce trust in that platform. Typically, the device provides trusted measurement and reporting of attributes of the associated platform, which indicate the integrity of the platform. Also, most preferably, the device is tamper-resistant.

In accordance with a first aspect, the present invention provides computing apparatus comprising mounted on an assembly main processing means and main memory means, each being connected for communication with one or more other components on the assembly,

characterised by further comprising a trusted device mounted on the assembly and being connected for communications with one or more other components on the assembly, the trusted device being arranged to acquire a true value of an integrity metric of the computing apparatus.

As used herein for reasons of simplicity of description, the term "device" also encompasses plural devices having equivalent function, or equivalent functionality integrated into one or more existing platform devices or assemblies. Additionally, the term 'true' as used herein implies that the value is that which correctly reflects the state of the computing apparatus. This may be ensured if the measurement method is substantially un-modifiable other than by the trusted device.

In accordance with a second aspect, the present invention provides a method of operating a system comprising trusted computing apparatus and a user, the trusted computing apparatus incorporating a trusted device being arranged to acquire the true value of an integrity metric of the computing apparatus, the method comprising the steps of:

the trusted device acquiring the true value of the integrity metric of the trusted computing apparatus;

the user generating a challenge for the trusted computing apparatus to prove its integrity and submitting the challenge to the trusted computing apparatus;

the trusted computing apparatus receiving the challenge, and the trusted device generating a response including the integrity metric and returning the response to the user; and

the user receiving the response, extracting the integrity metric from the response and comparing the integrity metric with an authenticated metric for the trusted computing apparatus that had been generated by a trusted party.

In accordance with a third aspect, the present invention provides a method of establishing a communications channel in a system between trusted computing apparatus and remote computing apparatus, the method including the step of the remote computing apparatus verifying the integrity of the trusted computing apparatus using the above method, and maintaining the communications channel for further transactions in the event the integrity of the trusted computing apparatus is successfully verified by the remote computing apparatus.

In accordance with a fourth embodiment, the present invention provides a method of verifying that trusted computing apparatus is trustworthy for use by a user for processing a particular application, the method including the step of the user verifying the integrity of the trusted computing apparatus using the above method, and the user using the trusted computing apparatus to process the particular application in the event the integrity of the trusted computing apparatus is successfully verified by the remote computing apparatus.

Other aspects and embodiments of the present invention will become apparent from the following description and claims.

Brief Description of the Drawings

A preferred embodiment of the present invention will now be described by way of example only with reference to the accompanying drawings in which:

Figure 1 is a diagram which shows the motherboard of computing apparatus adapted to include a trusted device according to an embodiment of the present invention;

Figure 2 is a diagram which shows in more detail the trusted device shown in Figure 1;

Figure 3 is a diagram which shows in the contents of a certificate stored in the trusted device;

Figure 4 is a diagram, which shows the features of a measurement function responsible for acquiring an integrity metric;

Figure 5 is a flow diagram which illustrates the steps involved in acquiring an integrity metric of the computing apparatus; and

Figure 6 is a flow diagram which illustrates the steps involved in establishing communications between a trusted computing platform and a remote platform including the trusted platform verifying its integrity.

Best Mode For Carrying Out the Invention, & Industrial Applicability

The present exemplary embodiment generally provides the incorporation into a computing platform of a physical trusted device whose function is to bind the identity of the platform to reliably measured data that provides an integrity metric of the platform. The identity and the integrity metric are compared with expected values provided by a trusted party (TP) that is prepared to vouch for the trustworthiness of the platform. If there is a match, the implication is that at least part of the platform is operating correctly, depending on the scope of the integrity metric.

A user verifies the correct operation of the platform before exchanging other data with the platform. A user does this by requesting the trusted device to provide its identity and an integrity metric. (Optionally the trusted device will refuse to provide evidence of identity if it itself was unable to verify correct operation of the platform.) The user receives the proof of identity and the identity metric, and compares them against values which it believes to be true. Those proper values are provided by the TP or another entity that is trusted by the user. If data reported by the trusted device is the same as that provided by the TP, the user trusts the platform. This is because the user trusts the entity. The entity trusts the platform because it has previously validated the identity and determined the proper integrity metric of the platform.

Once a user has established trusted operation of the platform, he exchanges other data with the platform. For a local user, the exchange might be by interacting with some software application running on the platform. For a remote user, the exchange might involve a secure transaction. In either case, the data exchanged is 'signed' by the trusted device. The user can then have greater confidence that data is being exchanged with a platform whose behaviour can be trusted.

The trusted device uses cryptographic processes but does not necessarily provide an external interface to those cryptographic processes. Also, a most desirable implementation would be to make the trusted device tamperproof, to protect secrets by making them inaccessible to other platform functions and provide an environment that is substantially immune to unauthorised modification. Since tamper-proofing is impossible, the best approximation is a trusted device that is tamper-resistant, or tamper-detecting. The trusted device, therefore, preferably consists of one physical component that is tamper-resistant.

Techniques relevant to tamper-resistance are well known to those skilled in the art of security. These techniques include methods for resisting tampering, methods for detecting tampering, and methods for eliminating data when tampering is detected. It will be appreciated that, although tamper-proofing is a most desirable feature of the present invention, it does not enter into the normal operation of the invention and, as such, is beyond the scope of the present invention and will not be described in any detail herein.

The trusted device is preferably a physical one because it must be difficult to forge. It is most preferably tamper-resistant because it must be hard to counterfeit. It typically has an engine capable of using cryptographic processes because it is required to prove identity, both locally and at a distance, and it contains at least one method of measuring some integrity metric of the platform with which it is associated.

Figure 1 illustrates the motherboard 10 of an exemplary computer platform (not shown). The motherboard 10 includes (among other standard components) a main processor 11, main memory 12, a trusted device 14, a data bus 16 and respective standard control lines 17 and address lines 18, and BIOS memory 19 containing the BIOS program for the platform.

Typically, the BIOS program is located in a special reserved memory area, the upper 64K of the first megabyte do the system memory (addresses FØØØh to FFFFh), and the main processor is arranged to look at this memory location first, in accordance with an industry wide standard

The significant difference between the platform and a conventional platform is that, after reset, the main processor is initially controlled by the trusted device, which then hands control over to the platform-specific BIOS program, which in turn initialises all input/output devices as normal. After the BIOS program has executed, control is handed over as normal by the BIOS program to an operating system program, such as Windows NT (TM), which is typically loaded into main memory 12 from a hard disk drive (not shown).

Clearly, this change from the normal procedure requires a modification to the implementation of the industry standard, whereby the main processor 11 is directed to address the trusted device 14 to receive its first instructions. This change may be made simply by hard-coding a different address into the main processor 11. Alternatively, the trusted device 14 may be assigned the standard BIOS program address, in which case there is no need to modify the main processor configuration.

Although, in the preferred embodiment to be described, the trusted device 14 is a single, discrete component, it is envisaged that the functions of the trusted device 14 may alternatively be split into multiple devices on the motherboard, or even integrated into one or more of the existing standard devices of the platform. For example, it is feasible to integrate one or more of the functions of the trusted device into the main processor itself, provided that the functions and their communications cannot be subverted. This, however, would probably require separate leads on the processor for sole use by the trusted functions. Additionally or alternatively, although in the present embodiment the trusted device is a hardware device that is adapted for integration into the motherboard 10, it is anticipated that a trusted device may be implemented as a 'removable' device, such as a dongle, which could be attached to a platform when required. Whether the trusted device is integrated or removable is a matter of design choice.

The trusted device 14 comprises a number of blocks, as illustrated in Figure 2: a controller 20 for controlling the overall operation of the trusted device 14, and interacting with

the other functions on the trusted device 14 and with the other devices on the motherboard 10; a measurement function 21 for acquiring an integrity metric from the platform; a cryptographic function 22 for signing or encrypting specified data; and interface circuitry 23 having appropriate ports (24, 25 & 26) for connecting the trusted device 14 respectively to the data bus 16, control lines 17 and address lines 18 of the motherboard 10. Each of the blocks in the trusted device 14 has access (typically via the controller 20) to appropriate volatile memory areas 27 and/or non-volatile memory areas 28 of the trusted device 14.

For reasons of performance, the trusted device 14 may be implemented as an application specific integrated circuit (ASIC). However, for flexibility, the trusted device is preferably an appropriately programmed micro-controller. Both ASICs and micro-controllers are well known in the art of microelectronics and will not be considered herein in any further detail.

One item of data stored in the non-volatile memory is a certificate 30, which is illustrated in Figure 3. The certificate 30 contains at least a public key 32 of the trusted device 14 and an authenticated value of a platform integrity metric 34 measured by a TP. Optionally, the trusted device 14 also contains an identity (ID) label 36 of the trusted device 14.

Where present, the ID label 36 is a conventional ID label, for example a serial number, that is unique within some context. The ID label 36 is generally used for indexing and labelling of data relevant to the trusted device 14, but is insufficient in itself to prove the identity of the platform under trusted conditions.

The trusted device 14 is equipped with at least one method of reliably measuring some integrity metric of the computing platform with which it is associated. The integrity metric is acquired by the measurement function 21, which is illustrated in more detail in Figure 4.

The measurement function 21 has access to non-volatile memory 40 for storing a hash program 41, plus volatile memory 42 for storing a computed integrity metric 43, in the form of a digest. The hash program 41 contains instructions for computing the digest, in code that is native to the main processor 11. In addition, part of the measurement function 21 is configured to respond to the main processor 11 as if it were addressable memory, such as standard read-only memory, by sensing memory read signals addressed to the trusted device 14 and returning appropriate data. The result is that the main processor 11 sees the trusted device, for the purposes of integrity metric measurement, as a standard read-only memory.

In the preferred implementation, as well as the digest, the integrity metric includes a Boolean value 44, which is stored in volatile memory 45 by the measurement function 21, for reasons that will become apparent.

A preferred process for acquiring an integrity metric will now be described with reference to Figure 5.

In step 500, at switch-on, the measurement function 21 monitors the activity of the main processor 11 on the data, control and address lines (16, 17 & 18) to determine whether the trusted device 14 is the first memory accessed. Under conventional operation, a main processor would first be directed to the BIOS memory first in order to execute the BIOS program. However, in accordance with the present embodiment, the main processor 11 is directed to the trusted device 14, which acts as a memory. In step 505, if the trusted device 14 is the first memory accessed, in step 510, the measurement function 21 writes to non-volatile memory 45 a Boolean value 44, which indicates that the trusted device 14 was the first memory accessed. Otherwise, in step 515, the measurement function writes a Boolean value 44, which indicates that the trusted device 14 was not the first memory accessed.

In the event the trusted device 14 is not the first accessed, there is of course a chance that the trusted device 14 will not be accessed at all. This would be the case, for example, if the main processor 11 were manipulated to run the BIOS program first. Under these circumstances, the platform would operate, but would be unable to verify its integrity on demand, since the integrity metric would not be available. Further, if the trusted device 14 were accessed after the BIOS program had been accessed, the Boolean value 44 would clearly indicate lack of integrity of the platform.

In step 520, when (or if) accessed as a memory by the main processor 11, the main processor 11 reads the stored native hash instructions 41 from the measurement function 21 in step 525. The hash instructions 41 are passed for processing by the main processor 11 over the data bus 16. In step 530, main processor 11 executes the hash instructions 41 and uses them, in step 535, to compute a digest of the BIOS memory 19, by reading the contents of the BIOS memory 19 and processing those contents according to the hash program. In step 540, the main processor 11 writes the computed digest 43 to the appropriate non-volatile memory location 42 in the trusted device 14. The measurement function 21, in step 545, then calls the BIOS program in the BIOS memory 19, and execution continues in a conventional manner.

Clearly, there are a number of different ways in which the integrity metric may be calculated, depending upon the scope of the trust required. The measurement of the BIOS program's integrity provides a fundamental check on the integrity of a platform's underlying

processing environment. Other integrity checks could involve establishing that various other devices, components or apparatus attached to the platform are present and in correct working order. In one example, the BIOS programs associated with a SCSI controller could be verified to ensure communications with peripheral equipment could be trusted. In another example, the integrity of other devices, for example memory devices or co-processors, on the platform could be verified by enacting fixed challenge/response interactions to ensure consistent results. Also, although in the present embodiment the trusted device 14 utilises the data bus as its main means of communication with other parts of the platform, it would be feasible, although not so convenient, to provide alternative communications paths, such as hard-wired paths or optical paths. Further, although in the present embodiment the trusted device 14 instructs the main processor 11 to calculate the integrity metric, it is anticipated that, in other embodiments, the trusted device itself will be arranged to measure one or more integrity metrics.

Preferably, the BIOS boot process includes mechanisms to verify the integrity of the boot process itself. Such mechanisms are already known from, for example, Intel's draft "Wired for Management baseline specification v 2.0 - BOOT Integrity Service", and involve calculating digests of software or firmware before loading that software or firmware. Such a computed digest is compared with a value stored in a certificate provided by a trusted entity, whose public key is known to the BIOS. The software/firmware is then loaded only if the computed value matches the expected value from the certificate, and the certificate has been proven valid by use of the trusted entity's public key. Otherwise, an appropriate exception handling routine is invoked.

Optionally, after receiving the computed BIOS digest, the trusted device 14 may inspect the proper value of the BIOS digest in the certificate and not pass control to the BIOS if the computed digest does not match the proper value. Additionally, or alternatively, the trusted device 14 may inspect the Boolean value 44 and not pass control back to the BIOS if the trusted device 14 was not the first memory accessed.

Figure 6 illustrates the flow of actions by a TP, the trusted device 14 incorporated into a platform, and a user (of a remote platform) who wants to verify the integrity of the trusted platform. It will be appreciated that substantially the same steps as are depicted in Figure 6 are involved when the user is a local user. In either case, the user would typically rely on some form of software application to enact the verification. It would be possible to run the software application on the remote platform or the trusted platform. However, there is a chance that, even on the remote platform, the software application could be subverted in some way. Therefore, it is anticipated that, for a high level of integrity, the software

application would reside on a smart card of the user, who would insert the smart card into an appropriate reader for the purposes of verification.

At the first instance, a TP, which vouches for trusted platforms, will inspect the type of the platform to decide whether to vouch for it or not. This will be a matter of policy. If all is well, in step 600, the TP measures the value of integrity metric of the platform. Then, the TP generates a certificate, in step 605, for the platform. The certificate is generated by the TP by appending the trusted device's public key, and optionally its ID label, to the measured integrity metric, and signing the string with the TP's private key.

The trusted device 14 can subsequently prove its identity by using its private key to process some input data received from the user and produce output data, such that the input/output pair is statistically impossible to produce without knowledge of the private key. Hence, knowledge of the private key forms the basis of identity in this case. Clearly, it would be feasible to use symmetric encryption to form the basis of identity. However, the disadvantage of using symmetric encryption is that the user would need to share his secret with the trusted device. Further, as a result of the need to share the secret with the user, while symmetric encryption would in principle be sufficient to prove identity to the user, it would insufficient to prove identity to a third party, who could not be entirely sure the verification originated from the trusted device or the user.

In step 610, the trusted device 14 is initialised by writing the certificate 30 into the appropriate non-volatile memory locations of the trusted device 14. This is done, preferably, by secure communication with the trusted device 14 after it is installed in the motherboard 10. The method of writing the certificate to the trusted device 14 is analogous to the method used to initialise smart cards by writing private keys thereto. The secure communications is supported by a 'master key', known only to the TP, that is written to the trusted device (or smart card) during manufacture, and used to enable the writing of data to the trusted device 14; writing of data to the trusted device 14 without knowledge of the master key is not possible.

At some later point during operation of the platform, for example when it is switched on or reset, in step 615, the trusted device 14 acquires and stores the integrity metric 42 of the platform.

When a user wishes to communicate with the platform, in step 620, he creates a nonce, such as a random number, and, in step 625, challenges the trusted device 14 (the operating system of the platform, or an appropriate software application, is arranged to recognise the challenge and pass it to the trusted device 14, typically via a BIOS-type call, in an appropriate fashion). The nonce is used to protect the user from deception caused by

replay of old but genuine signatures (called a 'replay attack') by untrustworthy platforms. The process of providing a nonce and verifying the response is an example of the well-known 'challenge/response' process.

In step 630, the trusted device 14 receives the challenge and creates a digest of the measured integrity metric and the nonce, and optionally its ID label. Then, in step 635, the trusted device 14 signs the digest, using its private key, and returns the signed digest, accompanied by the certificate 30, to the user.

In step 640, the user receives the challenge response and verifies the certificate using the well known public key of the TP. The user then, in step 650, extracts the trusted device's 14 public key from the certificate and uses it to decrypt the signed digest from the challenge response. Then, in step 660, the user verifies the nonce inside the challenge response. Next, in step 670, the user compares the computed integrity metric, which it extracts from the challenge response, with the proper platform integrity metric, which it extracts from the certificate. If any of the foregoing verification steps fails, in steps 645, 655, 665 or 675, the whole process ends in step 680 with no further communications taking place.

Assuming all is well, in steps 685 and 690, the user and the trusted platform use other protocols to set up secure communications for other data, where the data from the platform is preferably signed by the trusted device 14.

The techniques of signing, using certificates, and challenge/response, and using them to prove identity, are well known to those skilled in the art of security and will, thus, not be described in any more detail herein.

CLAIMS

1. A method for allowing a financial transaction to be performed using a electronic system, the method comprising interrogating an electronic transaction terminal with an electronic security device to obtain an integrity metric for the electronic financial transaction terminal; determining if the transaction terminal is a trusted terminal based upon the integrity metric; allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

2. A method according to claim 1, further comprising the providing of user identification data for the user of the electronic security data to the transaction terminal via the security device to allow authorisation of the transaction associated with the financial transaction data.

3. A financial transaction system comprising an electronic financial terminal; an electronic security device having interrogation means for interrogating the electronic financial transaction terminal to obtain an integrity metric for the electronic financial transaction terminal, determining means for determining if the transaction terminal is a trusted terminal based upon the integrity metric, means for allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

4. An electronic security transaction device having interrogation means for interrogating an electronic financial transaction terminal to obtain an integrity metric for the electronic financial transaction terminal, determining means for determining if the transaction

terminal is a trusted terminal based upon the integrity metric, means for allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

5

ABSTRACT

Trusted System

A method for allowing a financial transaction to be performed using a electronic system, the method comprising interrogating an electronic transaction terminal with an electronic security device to obtain an integrity metric for the electronic financial transaction terminal; determining if the transaction terminal is a trusted terminal based upon the integrity metric; allowing financial transaction data to be input into the transaction terminal if the transaction terminal is identified as a trusted terminal.

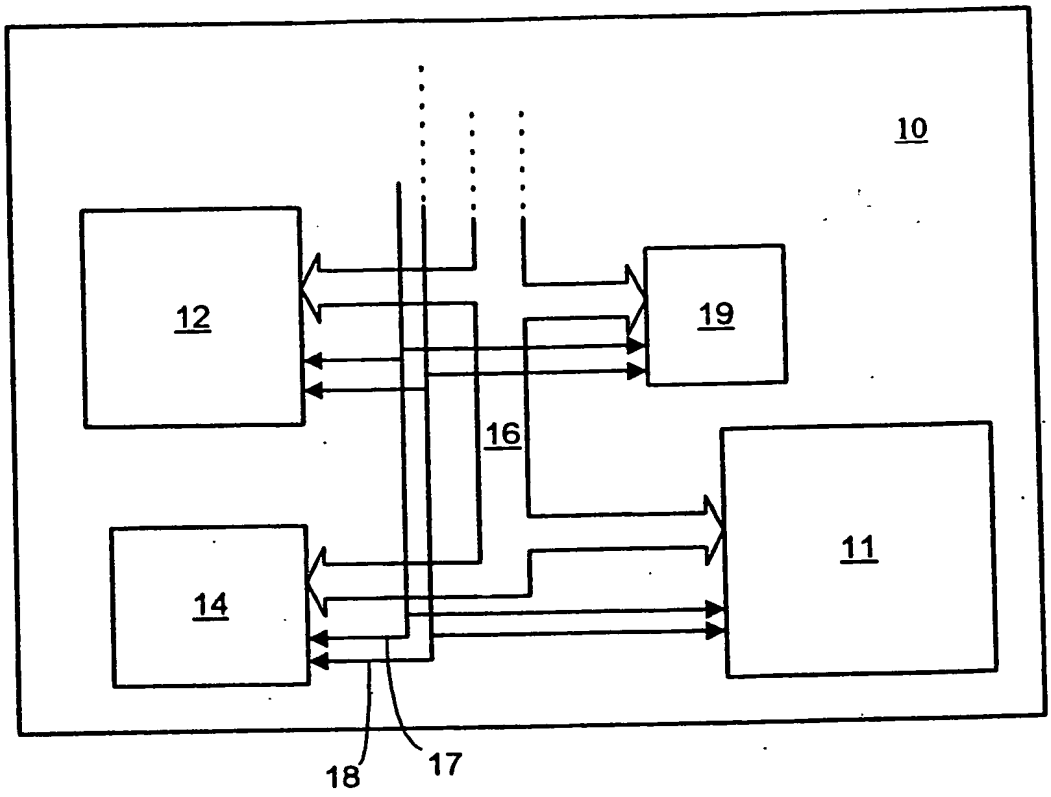


FIGURE 1

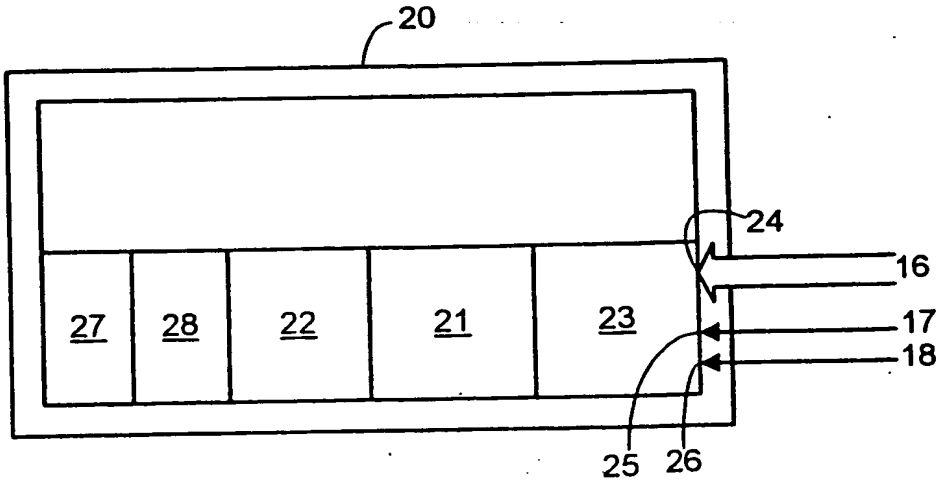


FIGURE 2

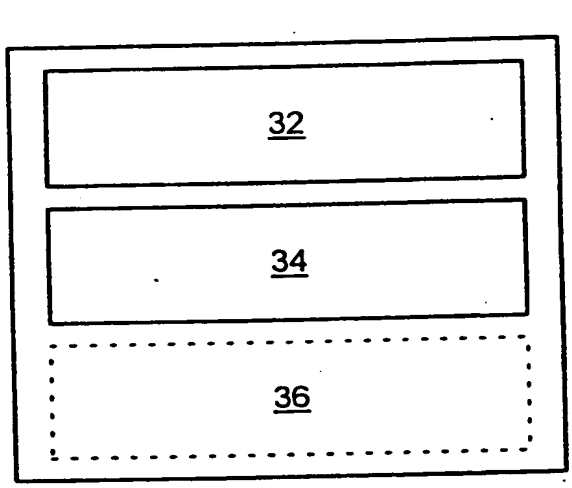


FIGURE 3

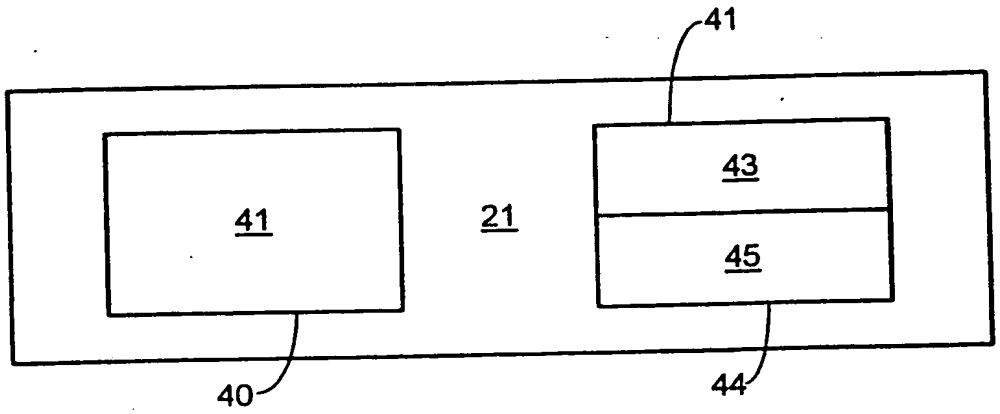


FIGURE 4

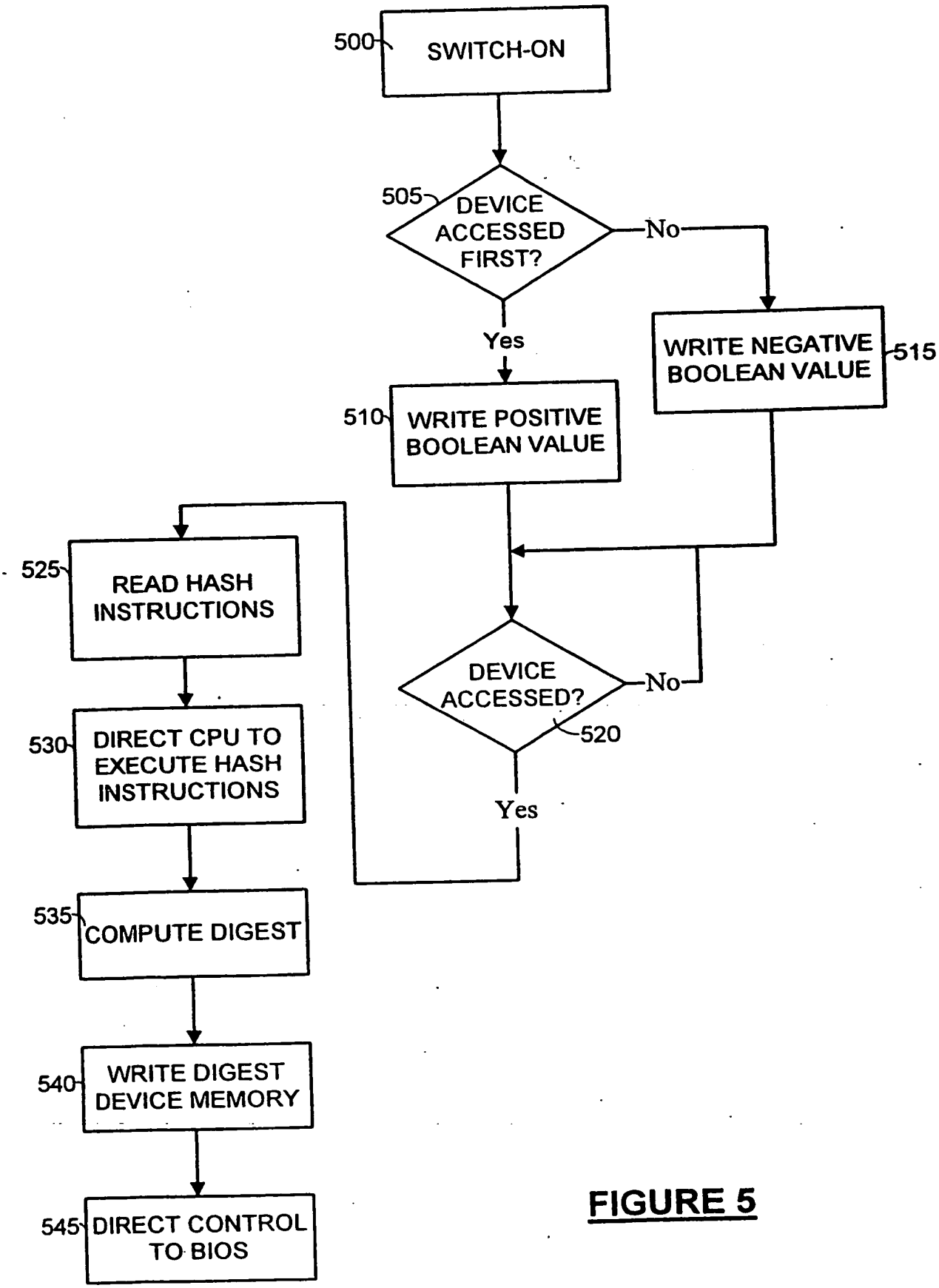

500
SWITCH-ON
505
DEVICE ACCESSED FIRST?
No
Yes
WRITE NEGATIVE BOOLEAN VALUE
515
510
WRITE POSITIVE BOOLEAN VALUE
525
READ HASH INSTRUCTIONS
DEVICE ACCESSED?
No
520
530
DIRECT CPU TO EXECUTE HASH INSTRUCTIONS
Yes
535
COMPUTE DIGEST
540
WRITE DIGEST DEVICE MEMORY
545
DIRECT CONTROL TO BIOS

FIGURE 5

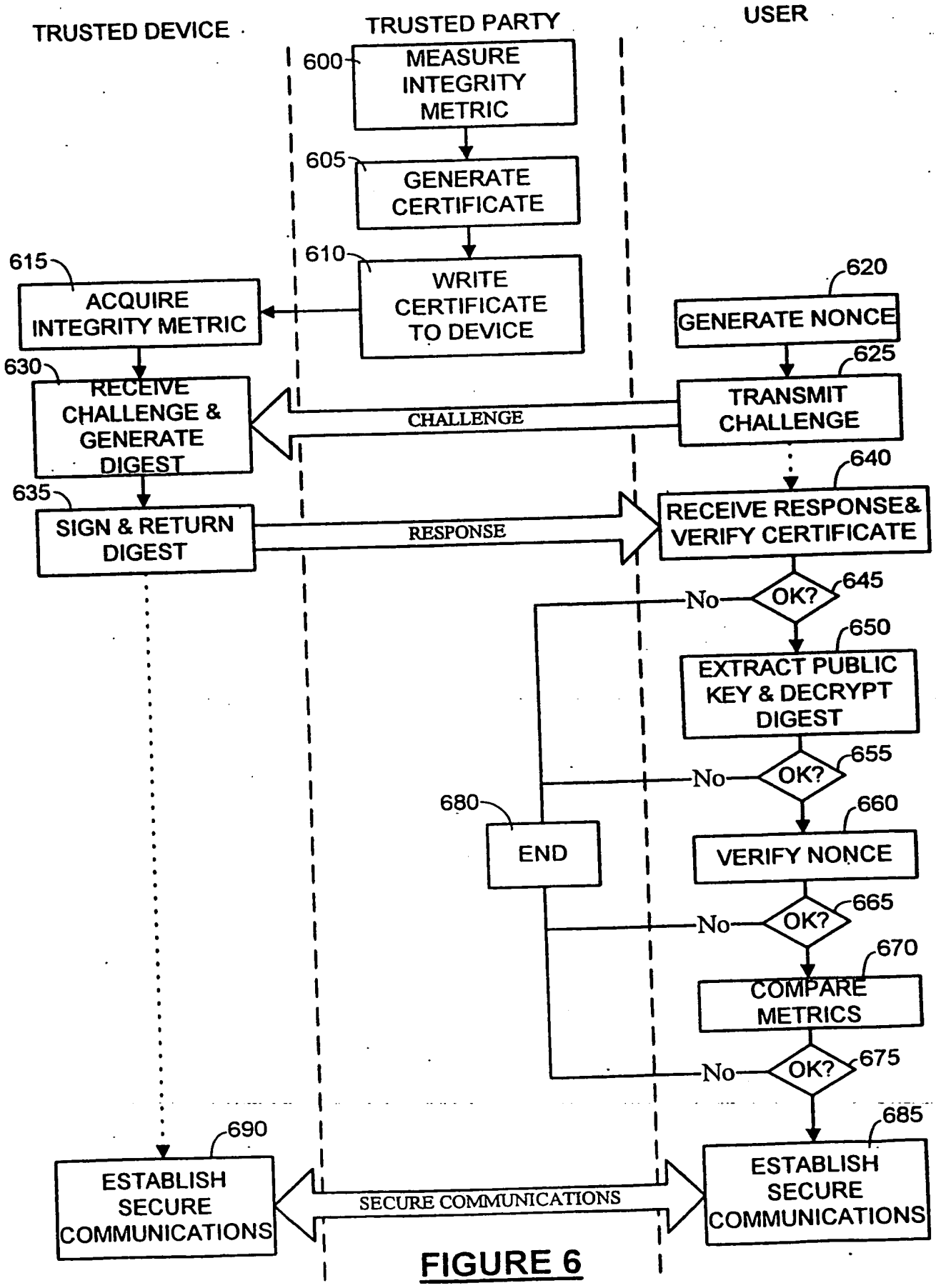

TRUSTED DEVICE
TRUSTED PARTY
USER
600
MEASURE INTEGRITY METRIC
605
GENERATE CERTIFICATE
610
WRITE CERTIFICATE TO DEVICE
615
ACQUIRE INTEGRITY METRIC
620
GENERATE NONCE
625
TRANSMIT CHALLENGE
CHALLENGE
630
RECEIVE CHALLENGE & GENERATE DIGEST
635
SIGN & RETURN DIGEST
RESPONSE
640
RECEIVE RESPONSE& VERIFY CERTIFICATE
645
OK?
No
650
EXTRACT PUBLIC KEY & DECRYPT DIGEST
655
OK?
No
660
VERIFY NONCE
665
OK?
No
670
COMPARE METRICS
675
OK?
No
680
END
685
ESTABLISH SECURE COMMUNICATIONS
690
ESTABLISH SECURE COMMUNICATIONS
SECURE COMMUNICATIONS

FIGURE 6